UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/709,415	05/04/2004	Brian Thinh-Vinh Tran	SVL920030099US1	3414

45727 7590 09/19/2006

LACASSE & ASSOCIATES, LLC
1725 DUKE STREET, SUITE 650
ALEXANDRIA, VA 22314

EXAMINER	
JOHNSON, JOHNESE T	
ART UNIT	**PAPER NUMBER**
2169	

DATE MAILED: 09/19/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	10/709,415	TRAN ET AL.
	Examiner	Art Unit
	Johnese Johnson	2191

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *04 May 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-13* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-13* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *04 May 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *12/07/2005*.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☒ Other: *See Continuation Sheet*.

Continuation of Attachment(s) 6). Other: IDS mailed on 06/09/2005 and 05/04/2004.

DETAILED ACTION

Claim Rejections - 35 USC § 112

1. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

2. Claims 1 is rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention.

Claim 1 is not supported by the specification. The figures do not show concatenation of local identifiers nor do they show any paths from a root node to a node, which is currently being assigned an identifier.

3. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

4. Claim 1 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

The limitation, "assigning to all subsequent nodes, node identifiers generated by a concatenation of local identifiers of all nodes along a path from a root node to a node to which a node identifier is currently being assigned", is not understandable to one of ordinary skill in the art. Since an identifier is a name, a name is not suppose to be a

value such as to be used in computation. A node itself may contain a value. Clarify the confusing usage of terms.

Claim Rejections - 35 USC § 103

5. Claims 1-4, and 9 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 6,889,226 issued to O'Neil et al (hereafter O'Neil), in view of U.S. Pat. No. 6,263,332 issued to Nasr et al (hereafter Nasr), and further in view of U.S. Pub No. 2002/0120679 issued to Hayton et al (hereafter, Hayton).

<u>Claim 1:</u>

Regarding claim 1, O'Neil discloses:

A method for prefix encoding node identifiers in a logical tree comprising steps of:

a. choosing an initial base length with which to encode local identifiers tree (see col. 5, lines 59-61).

c. sequentially assigning to descendants of a root node a local identifier having an even value and a length equal to said base length chosen in said choosing step, wherein said local identifiers are assigned in increasing value from leftmost children to rightmost children (see col. 15, line 9; wherein the nodes are sequentially numbered having an even value of 2),

e. extending said initial base length if local identifier encoding combinations are exhausted before all descendants are assigned local identifiers (see fig. 4; wherein the combinations are extended from 2 digits to 3 i.e. (1.5 to 1.5.1).

However, O'Neil does not disclose:

b. assigning a value of zero as a node identifier to a root node in a logical tree

d. assigning to all subsequent nodes, node identifiers generated by a concatenation of local identifiers of all nodes along a path from a root node to a node to which a node identifier is currently being assigned

Nasr discloses:

b. assigning a value of zero as a node identifier to a root node in a logical tree (see col. 5, line 57).

It would have been obvious, at the time of the invention, having the teachings of O'Neil and Nasr before him/her, to combine the step of assigning a value of zero as a node identifier to a root node in a logical tree as disclosed by Nasr with the steps of choosing an initial base length with which to encode local identifiers tree, sequentially assigning to descendants of a root node a local identifier having an even value and a length equal to said base length chosen in said choosing step, wherein said local identifiers are assigned in increasing value from leftmost children to rightmost children, and extending said initial base length if local identifier encoding combinations are exhausted before all descendants are assigned local identifiers as disclosed by O'Neil to designate the starting point for encoding succinct nodes and designate the highest level of the tree.

However, the combination of Nasr and O'Neil does not disclose:

d. assigning to all subsequent nodes, node identifiers generated by a concatenation of local identifiers of all nodes along a path from a root node to a node to which a node identifier is currently being assigned.

Hayton discloses:

d. assigning to all subsequent nodes, node identifiers generated by a concatenation of local identifiers of all nodes along a path from a root node to a node to which a node identifier is currently being assigned (see paragraph [0052], lines 3-9).

It would have been obvious, at the time of the invention, having the teachings of O'Neil, Nasr, and Hayton before him/her, to combine the step of assigning to all subsequent nodes, node identifiers generated by a concatenation of local identifiers of all nodes along a path from a root node to a node to which a node identifier is currently being assigned as disclosed by Hayton with the step of assigning a value of zero as a node identifier to a root node in a logical tree as disclosed by Nasr with the steps of choosing an initial base length with which to encode local identifiers tree, sequentially assigning to descendants of a root node a local identifier having an even value and a length equal to said base length chosen in said choosing step, wherein said local identifiers are assigned in increasing value from leftmost children to rightmost children, and extending said initial base length if local identifier encoding combinations are exhausted before all descendants are assigned local identifiers as disclosed by O'Neil to identify a nodes location relative to existing nodes in the tree.

Claim 2:

Regarding claim 2, O'Neil also discloses:

wherein inserting a node into an existing tree does not require change to existing node

identifiers (see paragraph [0050], lines 16-19, and 1-11; wherein a node is inserted between nodes after a tree has been constructed (existing tree) and only assigns the inserted node an identifier (does not require change)).

Claim 3:

Regarding claim 3, O'Neil also discloses:

wherein a node is inserted between a first node and a second node having consecutive local identifiers ([0049]14-15).

Claim 4:

Regarding claim 4, O'Neil discloses:

wherein said inserted node is assigned a local identifier having a string length longer than string length of said first node (see paragraph [0050], lines 27-29; wherein node 610 has a longer string length than node 608).

6. Claims 5-8 and 10-13 are rejected under 35 U.S.C. 103(a) as being unpatentable over of U.S. Pub No. 20030110150 issued to O'Neil et al (hereafter O'Neil), in view of U.S. Pub No. 2002/0120679 issued to Hayton et al (hereafter, Hayton).

Claim 5:

Regarding claim 5, claim 5 is rejected based upon the same reasoning as claim 1.

O'Neil also discloses:

wherein assigning said node identifier to an inserted node comprises the following steps:

a. determining whether node to be inserted is inserted as a first child, between two existing siblings, or as a last child under a single parent node (see col. 1, line 67 and col. 2, line 1; wherein before a node can be placed into a tree, the position has to be determined whether as a first child, between siblings, or as a last child),

b. if said node to be inserted is inserted as a first child under said single parent node (see col. 1, line 67 and col. 2, line 1; wherein before a node can be placed into a tree, the position has to be determined i.e. as a first child),

i. checking last byte of an existing first child (see col. 13, lines 45-49),

ii. if the value of said last byte is not the smallest even number, then an even number greater than zero and less than the value of said last byte is selected to generate a local identifier of said node to be inserted (see col. 15, lines 8-9), else

iii. if the value of said last byte of an existing first child is the smallest even number, generating a local identifier for said node to be inserted by replacing said last byte of said existing first child by an odd number to generate a local identifier and extending node identifier of said existing first child by a byte having a value of any arbitrary even number (see col. 15, lines 15-24; wherein an odd / even scheme is used/ discussed but the opposite scheme can be used),

c. if said node to be inserted is inserted between two existing siblings under said single parent node, determining whether the string length of node identifier of said first sibling is less than, equal to, or greater than the string length of node identifier of said second sibling (see col. 6, lines 58-67; wherein the length is determined before it is assigned), else

d. if said node to be inserted is inserted as a last child after all other children under said single parent node, assigning to said node to be inserted an even local identifier greater than that of existing last child under said single parent node (see fig. 6 wherein the inserted node (606) has a greater identifier than its parent (602)).

However, O'Neil does not disclose:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

Hayton discloses:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted (see paragraph [0052], lines 3-9).

It would have been obvious, at the time of the invention, having the teachings of O'Neil, and Hayton before him/her, to combine the step of generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted as disclosed by Hayton with the steps as disclosed by O'Neil to define a path through the components of the application to a property at the end of the concatenation, and to associate the element of the user-interface with the current state of that property (see paragraph [0011], lines 9-14).

Claim 6:

Regarding claim 6, claim 6 is rejected on the same basis of claim 5.

O'Neil also discloses:

a. checking if local identifier of said first sibling is the last available encoding value having a string length of the local identifier of said first sibling and being smaller in value than said local identifier of said second sibling (see figure 6 wherein before assigning an identifier to node 606, both identifiers are checked and node 606 receives an identifier value that's between the values of both siblings),

b. if said local identifier of said first sibling is the last combination having a string length of the local identifier of said first sibling that is smaller in value than said local identifier of said second sibling (see col. 6, lines 58-67; wherein the length is checked before it is assigned),

i. if the local identifier of said second sibling is not the first available identifier having the string length of the local identifier of said second sibling that is greater than the value of said local identifier of said first sibling; an even-valued local identifier being less in value than said local identifier of said second sibling and having string length of local identifier of said second sibling is generated and assigned (see col. 15, lines 15-24;), else

ii. generating a local identifier for said node to be inserted by replacing said last byte of said existing first child by an odd number and extending local identifier of said existing first child by a byte having a value of any arbitrary even number less in value than said last byte of said existing first child (see col. 15, lines 15-24), and

However, O'Neil does not disclose:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

Hayton discloses:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted (see paragraph [0052], lines 3-9).

It would have been obvious, at the time of the invention, having the teachings of O'Neil, and Hayton before him/her, to combine the step of generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted as disclosed by Hayton with the steps as disclosed by O'Neil to define a path through the components of the application to a property at the end of the concatenation, and to associate the element of the user-interface with the current state of that property (see paragraph [0011], lines 9-14).

Claim 7:

Regarding claim 7, claim 7 is rejected on the same basis as claim 5.

O'Neil also discloses:

a. if the value of the local identifier of said first sibling plus two is less than the value of the local identifier of said second sibling, a local identifier for said node to be inserted takes on an even value greater than or equal to the value of said local identifier of first sibling plus two and less than the value of the local identifier of said second sibling (see col. 9, lines 15-24 and 27-37; wherein the reference discloses a numbering

scheme divisible by three but states that any scheme, that obeys the properties, can be used.)

b. if the string length of the local identifier of said first sibling plus two is equal to the string length of the local identifier of said second sibling, then the string length of the local identifier for said node to be inserted is extended wherein the length of the local identifier for the newly inserted node is the string length of said second sibling plus one, and the value of the first string length of said first sibling bytes is the node identifier of said first sibling plus one, and the new byte is an arbitrary even number less than the value of said last byte of the node identifier of said second sibling, and generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted (see col. 9, lines 15-24 and 27-37; wherein the reference discloses a numbering scheme divisible by three but states that any scheme, that obeys the properties, can be used.).

However, O'Neil does not disclose:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

Hayton discloses:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted (see paragraph [0052], lines 3-9).

It would have been obvious, at the time of the invention, having the teachings of O'Neil, and Hayton before him/her, to combine the step of generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said

node to be inserted as disclosed by Hayton with the steps as disclosed by O'Neil to define a path through the components of the application to a property at the end of the concatenation, and to associate the element of the user-interface with the current state of that property (see paragraph [0011], lines 9-14).

Claim 8:

Regarding claim 8, claim 8 is rejected on the same basis as claim 5.

O'Neil also discloses:

a. if the local identifier of said second sibling is not the smallest value having the string length of said second sibling that is greater in value than the local identifier of said first sibling, then a local identifier having a string length of said second sibling and having even value smaller than the value of the last byte of the node identifier of said second sibling is generated and assigned else (see col. 9, lines 15-24 and 27-37; wherein the reference discloses a numbering scheme divisible by three but states that any scheme, that obeys the properties, can be used; also, see fig. 6 wherein the inserted node (606) has a smaller value than that of the second sibling),

b. if the local identifier of said first sibling is not the largest value with the string length of the local identifier of said first sibling, one of the larger values for the new encoding is generated and assigned (see fig. 6 wherein a larger value for the inserted node is generated and assigned), else

c. extending the local identifier of said first sibling by a length, by setting the last byte to the highest odd number and the new byte to an even number less than the value of the last byte, and generating a node identifier by a concatenation of local identifiers of all

nodes along a path from a root node to said node to be inserted (see fig.6 wherein the identifier of the inserted node (606) is extended by a length).

However, O'Neil does not disclose:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

Hayton discloses:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted (see paragraph [0052], lines 3-9).

It would have been obvious, at the time of the invention, having the teachings of O'Neil, and Hayton before him/her, to combine the step of generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted as disclosed by Hayton with the steps as disclosed by O'Neil to define a path through the components of the application to a property at the end of the concatenation, and to associate the element of the user-interface with the current state of that property (see paragraph [0011], lines 9-14).

Claim 9:

Regarding claim 9, O'Neil discloses:

An article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements prefix encoding node identifiers in a logical tree comprising modules implementing code for (see col. 3, lines 17-29):

a. choosing an initial base length with which to encode local identifiers tree (see col. 5, lines 59-61).

c. sequentially assigning to descendants of a root node a local identifier having an even value and a length equal to said base length chosen in said choosing step, wherein said local identifiers are assigned in increasing value from leftmost children to rightmost children (see col. 15, line 9; wherein the nodes are sequentially numbered having an even value of 2),

e. extending said initial base length if local identifier encoding combinations are exhausted before all descendants are assigned local identifiers (see fig. 4; wherein the combinations are extended from 2 digits to 3 i.e. (1.5 to 1.5.1).

However, O'Neil does not disclose:

b. assigning a value of zero as a node identifier to a root node in a logical tree

d. assigning to all subsequent nodes, node identifiers generated by a concatenation of local identifiers of all nodes along a path from a root node to a node to which a node identifier is currently being assigned

Nasr discloses:

b. assigning a value of zero as a node identifier to a root node in a logical tree (see col. 5, line 57).

It would have been obvious, at the time of the invention, having the teachings of O'Neil and Nasr before him/her, to combine the step of assigning a value of zero as a node identifier to a root node in a logical tree as disclosed by Nasr with the steps of choosing an initial base length with which to encode local identifiers tree, sequentially

assigning to descendants of a root node a local identifier having an even value and a length equal to said base length chosen in said choosing step, wherein said local identifiers are assigned in increasing value from leftmost children to rightmost children, and extending said initial base length if local identifier encoding combinations are exhausted before all descendants are assigned local identifiers as disclosed by O'Neil to designate the starting point for encoding succinct nodes and designate the highest level of the tree.

However, the combination of Nasr and O'Neil does not disclose:

d. assigning to all subsequent nodes, node identifiers generated by a concatenation of local identifiers of all nodes along a path from a root node to a node to which a node identifier is currently being assigned.

Hayton discloses:

d. assigning to all subsequent nodes, node identifiers generated by a concatenation of local identifiers of all nodes along a path from a root node to a node to which a node identifier is currently being assigned (see paragraph [0052], lines 3-9).

It would have been obvious, at the time of the invention, having the teachings of O'Neil, Nasr, and Hayton before him/her, to combine the step of assigning to all subsequent nodes, node identifiers generated by a concatenation of local identifiers of all nodes along a path from a root node to a node to which a node identifier is currently being assigned as disclosed by Hayton with the step of assigning a value of zero as a node identifier to a root node in a logical tree as disclosed by Nasr with the steps of choosing an initial base length with which to encode local identifiers tree, sequentially

assigning to descendants of a root node a local identifier having an even value and a length equal to said base length chosen in said choosing step, wherein said local identifiers are assigned in increasing value from leftmost children to rightmost children, and extending said initial base length if local identifier encoding combinations are exhausted before all descendants are assigned local identifiers and article of manufacture as disclosed by O'Neil to identify a nodes location relative to existing nodes in the tree.

Claim 10:

Regarding claim 10, claim 10 is rejected based upon the same reasoning as claim 9.

O'Neil also discloses:

wherein assigning a prefix encoded node identifier to an inserted node comprises modules implementing code (see col. 3, line 27) for:

a. determining whether node to be inserted is inserted as a first child, between two existing siblings, or as a last child under a single parent node (see col. 1, line 67 and col. 2, line 1; wherein before a node can be placed into a tree, the position has to be determined whether as a first child, between siblings, or as a last child),

b. if said node to be inserted is inserted as a first child under said single parent node (see col. 1, line 67 and col. 2, line 1; wherein before a node can be placed into a tree, the position has to be determined i.e. as a first child),

i. checking last byte of an existing first child (see col. 13, lines 45-49),

ii. if the value of said last byte is not the smallest even number, then an even

number greater than zero and less than the value of said last byte is selected to generate a local identifier of said node to be inserted (see col. 15, lines 8-9), else

iii. if the value of said last byte of an existing first child is the smallest even number, generating a local identifier for said node to be inserted by replacing said last byte of said existing first child by an odd number to generate a local identifier and extending node identifier of said existing first child by a byte having a value of any arbitrary even number (see col. 15, lines 15-24; wherein an odd / even scheme is used/ discussed but the opposite scheme can be used),

c. if said node to be inserted is inserted between two existing siblings under said single parent node, determining whether the string length of node identifier of said first sibling is less than, equal to, or greater than the string length of node identifier of said second sibling (see col. 6, lines 58-67; wherein the length is determined before it is assigned), else

d. if said node to be inserted is inserted as a last child after all other children under said single parent node, assigning to said node to be inserted an even local identifier greater than that of existing last child under said single parent node (see fig. 6 wherein the inserted node (606) has a greater identifier than its parent (602)).

However, O'Neil does not disclose:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

Hayton discloses:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted (see paragraph [0052], lines 3-9).

It would have been obvious, at the time of the invention, having the teachings of O'Neil, and Hayton before him/her, to combine the step of generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted as disclosed by Hayton with the steps and article of manufacture as disclosed by O'Neil to define a path through the components of the application to a property at the end of the concatenation, and to associate the element of the user-interface with the current state of that property (see paragraph [0011], lines 9-14).

<u>Claim 11:</u>

Regarding claim 11, claim 11 is rejected on the same basis of claim 10.

O'Neil also discloses:

a. checking if local identifier of said first sibling is the last available encoding value having a string length of the local identifier of said first sibling and being smaller in value than said local identifier of said second sibling (see figure 6 wherein before assigning an identifier to node 606, both identifiers are checked and node 606 receives an identifier value that's between the values of both siblings),

b. if said local identifier of said

first sibling is the last combination having a string length of the local

identifier of said first sibling that is smaller in value than said local

identifier of said second sibling (see col. 6, lines 58-67; wherein the length is checked before it is assigned),

i. if the local identifier of said second sibling is not the first available identifier having the string length of the local identifier of said second sibling that is greater than the value of said local identifier of said first sibling; an even-valued local identifier being less in value than said local identifier of said second sibling and having string length of local identifier of said second sibling is generated and assigned (see col. 15, lines 15-24;), else

ii. generating a local identifier for said node to be inserted by replacing said last byte of said existing first child by an odd number and extending local identifier of said existing first child by a byte having a value of any arbitrary even number less in value than said last byte of said existing first child (see col. 15, lines 15-24), and

However, O'Neil does not disclose:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

Hayton discloses:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted (see paragraph [0052], lines 3-9).

It would have been obvious, at the time of the invention, having the teachings of O'Neil, and Hayton before him/her, to combine the step of generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted as disclosed by Hayton with the steps and article as disclosed by

O'Neil to define a path through the components of the application to a property at the end of the concatenation, and to associate the element of the user-interface with the current state of that property (see paragraph [0011], lines 9-14).

Claim 12:

Regarding claim 12, claim 12 is rejected on the same basis as claim 10.

O'Neil also discloses:

a. if the value of the local identifier of said first sibling plus two is less than the value of the local identifier of said second sibling, a local identifier for said node to be inserted takes on an even value greater than or equal to the value of said local identifier of first sibling plus two and less than the value of the local identifier of said second sibling (see col. 9, lines 15-24 and 27-37; wherein the reference discloses a numbering scheme divisible by three but states that any scheme, that obeys the properties, can be used.)

b. if the string length of the local identifier of said first sibling plus two is equal to the string length of the local identifier of said second sibling, then the string length of the local identifier for said node to be inserted is extended wherein the length of the local identifier for the newly inserted node is the string length of said second sibling plus one, and the value of the first string length of said first sibling bytes is the node identifier of said first sibling plus one, and the new byte is an arbitrary even number less than the value of said last byte of the node identifier of said second sibling, and generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted (see col. 9, lines 15-24 and 27-37; wherein the

reference discloses a numbering scheme divisible by three but states that any scheme, that obeys the properties, can be used.).

However, O'Neil does not disclose:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

Hayton discloses:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted (see paragraph [0052], lines 3-9).

It would have been obvious, at the time of the invention, having the teachings of O'Neil, and Hayton before him/her, to combine the step of generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted as disclosed by Hayton with the steps and article as disclosed by O'Neil to define a path through the components of the application to a property at the end of the concatenation, and to associate the element of the user-interface with the current state of that property (see paragraph [0011], lines 9-14).

Claim 13:

Regarding claim 13, claim 13 is rejected on the same basis as claim 10.

O'Neil also discloses:

a. if the local identifier of said second sibling is not the smallest value having the string length of said second sibling that is greater in value than the local identifier of said first sibling, then a local identifier having a string length of said second sibling and having even value smaller than the value of the last byte of the node identifier of said second

sibling is generated and assigned else (see col. 9, lines 15-24 and 27-37; wherein the reference discloses a numbering scheme divisible by three but states that any scheme, that obeys the properties, can be used; also, see fig. 6 wherein the inserted node (606) has a smaller value than that of the second sibling),

b. if the local identifier of said first sibling is not the largest value with the string length of the local identifier of said first sibling, one of the larger values for the new encoding is generated and assigned (see fig. 6 wherein a larger value for the inserted node is generated and assigned), else

c. extending the local identifier of said first sibling by a length, by setting the last byte to the highest odd number and the new byte to an even number less than the value of the last byte, and generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted (see fig.6 wherein the identifier of the inserted node (606) is extended by a length).

However, O'Neil does not disclose:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted.

Hayton discloses:

generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said node to be inserted (see paragraph [0052], lines 3-9).

It would have been obvious, at the time of the invention, having the teachings of O'Neil, and Hayton before him/her, to combine the step of generating a node identifier by a concatenation of local identifiers of all nodes along a path from a root node to said

node to be inserted as disclosed by Hayton with the steps and article as disclosed by O'Neil to define a path through the components of the application to a property at the end of the concatenation, and to associate the element of the user-interface with the current state of that property (see paragraph [0011], lines 9-14).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Johnese Johnson whose telephone number is 571-270-1097. The examiner can normally be reached on 4/5/9.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, David V. Bruce can be reached on 571-272-2487. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

2006 September 11, 2006

David Bruce

DAVID BRUCE
SUPERVISORY PATENT EXAMINER

JTJ
JTJ